UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/693,392	10/24/2003	Jeffrey P. Snover	MS1-1738US	1997

22801 7590 09/02/2008

LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA 99201

EXAMINER	
CAO, DIEM K	

ART UNIT	PAPER NUMBER
2194	

MAIL DATE	DELIVERY MODE
09/02/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/693,392	SNOVER ET AL.	
	Examiner	Art Unit	
	DIEM K. CAO	2194	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *11 June 2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-31 and 33-38* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-31,33-38* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-31 and 33-38 are pending. Applicant has amended claims 1—19, 28-29, 35-36 and 38 and canceled claim 32.

Continued Examination Under 37 CFR 1.114

2. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 6/11/2008 has been entered.

Claim Rejections - 35 USC § 112

3. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

4. Claims 1-18, 29-31 and 33-36 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Claim 1 recites "one input parameter from information included in an input source, the input source comprising at least one pipelined live object outputted by a second data structure

that is identical to the first data structure", however, the limitation was not described in the specification in such a way as to reasonably convey to one of skilled in the art that the invention, at the time of the application was filed, has possession of the claimed invention. The specification seems to disclose (See Fig. 13, and specification, page 46, line 13 – page 47, line 5) the input source comprises multiple parts, each parts is associated with one of the objects instantiated from multiple data structures/classes, and during execution of the objects, each object may output live object which is input as input object to the next pipelined object. Though, the input source does not include the object outputted from objects that instantiated from the data structures/classes.

Claims 2-18 fail to remedy the deficiencies of claim 1 above and also rejected under the same ground of rejection.

Claim 29 suffers the same problem as claim 1 above and is rejected under the same ground of rejection;

Claims 30-31 and 33-36 fail to remedy the deficiencies of claim 29 above and also rejected under the same ground of rejection.

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 1-31 and 33-38 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 1 recites "wherein the parameter definition is a declared property in a declared class of the first data structure", and "when the first data structure becomes instantiated into an object", however, the specification does not seem to disclose "a declared class of the first data structure", rather, the first data structure is the declared class itself. Furthermore, in the object oriented programming, an object is instantiated from a class, not a structure, therefore, claim 1 is rejected as being indefinite for failing to point out and distinctly claim the subject matter.

Claims 19, 28 and 29 suffer the same problem as claim 1 above, and are rejected under the same ground of rejection.

Claim 1 further recites "one pipelined live object outputted by a second data structure that is identical to the first data structure", which is unclear how it is supported by the specification. The specification seems to disclose the pipelined object is outputted by a second object instantiated from a second class/data structure. In addition, the claim recites "identical to the first data structure" which makes claim indefinite because "identical" does not seem to support by the specification. A first data structure is a class and a second data structure is also a class, but two classes are not identical.

Claim 15 recites "wherein the data structure is a public class" which is unclear as to which the data structure is referred to, i.e., "the first data structure" or "the second data structure". Furthermore, now claim 15, which depends on claim 1, claims "wherein the parameter definition is a declared property in a declared class of a public class", which is doesn't make sense at all.

The rest of the claims are depend on the parent claims and fail to remedy the deficiencies of parent claims above and are also rejected under the same ground of rejection

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. Claims 1-8, 10-18, 29-31, 33, 36 and 38 rejected under 35 U.S.C. 103(a) as being unpatentable over Little et al. (U.S. 6,907,572 B2) in view of Shalabi et al. (U.S. 2004/0143599 A1).

As to claim 1, Little teaches a computer readable medium encoded with a data structure, comprising:

- a parameter definition for at least one input parameter (private String [] argument_types; col. 8, lines 55-60), the parameter definition being configured to enable identification of an appropriate input for the at least one parameter (The Java class CLIAction translates arguments from the input string; col. 9, lines 16-18), wherein the parameter definition is created via a class declaration within the data structure (CLIAction class; col. 8, lines 55-56), and
- an instruction-based mechanism configured to use the parameter definition to identify the appropriate input for the at least one input parameter from information included in an input source (The Java class … then invokes the object and method required for that

input; col. 9, lines 16-18 and For each of the types in the … command line; col. 11, lines 14-22 and 26-44), the input source comprising at least one live object (command line; col. 11, lines 21-22),

- wherein the instruction-based mechanism is further configured to process the at least one input parameter based on the appropriate input to output a live object when the data structure becomes instantiated into an object (Upon receiving the serialized input … into the actual object, method and constructors that are used to invoke the method; col. 10, lines 53-67 and For each of the types in the … command line; col. 11, lines 14-22 and 26-44).

Little does not teach the input source comprising at least one pipelined live object outputted by a second data structure that is identical to the first data structure, apply one or more directives associated with the declared class that specify at least one of a machine role or a use role. However, Shalabi teaches the input source comprising at least one pipelined live object outputted by a second data structure that is identical to the first data structure (page 7, paragraph [0103], [0110], page 9, paragraphs [0163] and [0167]), apply one or more directives associated with the declared class that specify at least one of a machine role or a use role (page 6, paragraph [0085], page 7, paragraph [0097]).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Shalabi to the system of Little because Shalabi teaches a method for administering multiples places deployed on multiple servers, thus a user can control/utilizes multiple servers from a single user interface.

As to claim 2, Little teaches the input source comprises a string (command line; col. 11, lines 21-22).

As to claim 3, Little does not teach the string comprises a part of a script. However, Shalabi teaches string comprises a part of a script (batch file; page 9, paragraph [0163]).

As to claim 4, Little teaches the string comprises a part of a command string entered on a command line (command line; col. 11, lines 21-22).

As to claim 5, Little teaches wherein the parameter definition comprises a data type (String; col. 8, line 59) and a name for the expected input parameter (time, day of month, month and year; col. 7, lines 58-60).

As to claim 6, Little teaches wherein the information comprises a value (For example consider the day of the month argument for the "clock set" command … to this argument; col. 13, line 65 – col. 14, line 3).

As to claim 7, see rejection of claim 5 above. Little further teaches wherein the mechanism further coerces the value having a first data type into a converted value having a second data type specified in the definition (A string input to this argument that cannot be parsed into an integer is invalid; col. 13, line 67 – col. 14, line 1).

As to claim 8, Little teaches the input source comprises a set of objects (show, show clock details; col. 12, lines 15-20).

As to claim 10, Little teaches wherein the input source comprises a precisely parseable stream (A string input to this argument that cannot be parsed into an integer is invalid; col. 13, line 67 – col. 14, line 1).

As to claim 11, Little do not teach wherein the precisely parseable stream comprises an XML-based document. However, Shalabit teaches a parser parses and validates an XML document (page 7, paragraphs [0103]-[0104]).

As to claim 12, Little teaches wherein the mechanism further identifies and populates each expected input parameter for each record within the input source (For each of the types in the … command line; col. 11, lines 14-22 and 26-44).

As to claim 13, Little teaches a mapping mechanism that is operative to associate a mapped name with the input parameter, wherein identifying the information is based on the mapped name ("string name"; col. 9, lines 47-62 and col. 7, lines 17-52).

As to claim 14, Little teaches wherein the mechanism comprises a method inherited from a class provided within a runtime environment (method_name; col. 10, lines 55-67).

As to claim 15, Little teaches wherein the data structure is a public class (CLIAction class; col. 10, line 54 - col. 11, line 10).

As to claim 16, Little teaches wherein the input parameters are public parameters (col. 11, lines 33-44).

As to claim 17, Little teaches a plurality of parameter definitions, each parameter definition corresponding to one of a plurality of input parameters, wherein at least one of the parameter definitions is configured to be indirectly associated with the data structure (an <action> tag content in the XML CLI … a Java class CLIAction … private String [] argument_types; col. 8, lines 45-60 and This includes defining the Java class … mapping in XML; col. 11, lines 58-65).

As to claim 18, Little teaches wherein the indirect association between the at least one of the parameter definitions and the data structure comprises a reference to an XML based document that enables identification of the corresponding input parameter (XML CLI description files; see Figs. 2-3 and associated text).

As to claim 38, Little teaches wherein the live object is of a data type having a method, the method being directly invocable when processing the expected input parameter (see Fig. 20).

As to claim 29, Little teaches

- a name identifying an application that is included in a declared parent class provided by an object-based environment (CLI application; col. 3, lines 16-18, and col. 4, lines 6-7 and col. 7, lines 35-41),
- at least one member configured to receive one or more sets of input, wherein each set of input comprises at least one live object (When a user … CLI command … action; col. 5, lines 55-58 and token on the command line; col. 11, lines 20-21),
- a method associated with the one or more sets of input (method; col. 11, lines 14-15),
- wherein the declared parent class is configured to provide processing that executes the method for each set of input received for the at least one member when the name of the application is invoked (when the command "show clock" command … in the action; col. 5, lines 59-67).

Little does not explicitly teach the application comprises a command in a pipeline of commands, and the set of input comprises results from a previous command in the pipeline of commands. However, Shalabi teaches the application comprises a command in a pipeline of commands (The Administrator then issues a series of input commands; page 7, paragraph [0103]), and the set of input comprises results from a previous command in the pipeline of commands (Output of one command may be input to another; page 7, paragraph [0110], page 9, paragraphs [0163] and [0167]).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Shalabi to the system of Little because Shalabi teaches a method for administering multiples places deployed on multiple servers, thus a user can control/utilizes multiple servers from a single user interface.

As to claim 30, Little teaches wherein the at least one member comprises an expected input parameter (time, day of month, month and year; col. 7, lines 58-60).

As to claim 31, Little teaches wherein the declared parent class further provides validation processing on each set of input for the expected input parameter and does not execute the method for one set of input if the one set fails the validation processing (Argument validation … throwing an ArgumentException instance; col. 13, line 54 – col. 14, line 25).

As to claim 33, Little teaches wherein each set of input includes an identifier that associated the input with the member (object_name, method_name; col. 10, lines 60-61 and col. 12, lines 15-29).

As to claim 36, Little teaches wherein the command is invoked via an object-based command line environment (When a user … CLI command … action; col. 5, lines 55-58 and token on the command line; col. 11, lines 20-21).

9. Claims 19-23, 26, 28 and 37 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gillis et al. (U.S. 6,286,035 B1) in view of Little et al. (U.S. 6,907,572 B2).

As to claim 19, Gillis teaches a computer-executable method for populating parameters declared within a data structure (abstract), the method comprising:

obtaining an expected name for a parameter, the expected name being assigned in a declaration for the parameter within a data structure (determined whether an associated parameter description is found using the unique key of the received parameter; col. 7, lines 60-62),

identifying a label within an input source correlating to the expected name (the function finds the associated received parameter using the unique key of the parameter descriptor; col. 8, lines 17-20), the input source not comprising a live object (invalid command message; col. 7, lines 52-55),

retrieving a value associated with the label (the value of the received parameter; col. 8, lines 29-30), and

assigning the value to the parameter (the value of the received parameter ... message structure; col. 8, lines 35-43).

Gillis does not explicitly teach the expected name being assigned via a declared property in a declared class, applying one or more declared directives associated with the declared class of a data structure to the value, the one or more declared directives being associated with the parameter, wherein the one or more declared directive are configured to cause an administrative tool framework to process the parameter. However, Little teaches the expected name being assigned via a declared property in a declared class (time, day of month, month and year; col. 7, lines 58-60 and CLIAction class; col. 8, lines 50-60), applying one or more declared directives associated with the declared class of a data structure to the value, the one or more declared directives being associated with the parameter, wherein the one or more declared directive are configured to cause an administrative tool framework to process the parameter (the runtime

system performs object mapping, method mapping and the arguments are provided to carry out the action; col. 5, lines 55-67, col. 10, lines 53-67 and validating the arguments and performing the required action; col. 13, line 54 - col. 14, line 20).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Little to the system of Gillis because Little teaches a method that maps command to objects and method defined in Java classes, and using this method will shield users from changes or version upgrades applied to the system (col. 14, lines 31-44).

As to claim 20, Gillis teaches wherein the expected name and label are identical (unique key; col. 7, lines 60-63).

As to claim 21, Gillis teaches providing mapping information that defines an alias name for the expected name and identifying the label based on the alias name (unique key; col. 7, lines 60-63).

As to claim 22, Gillis teaches wherein the input source comprises a command string entered on a command line (col. 1, lines 29-33) and the alias name is provided within the command string (unique key; col. 7, lines 60-63).

As to claim 23, Gillis teaches wherein the alias name is provided within a data store (the function finds the associated received parameter using the unique key of the parameter descriptor; col. 8, lines 17-20).

As to claim 26, Gillis teaches wherein the input source comprises a command string entered on a command line (col. 1, lines 29-33).

As to claim 37, Gillis teaches validating the value and wherein assigning the value to the parameter occurs if the value passes the validation (col. 8, lines 28-43).

As to claim 28, it is the same as the method claim 19 except it is a computer system claim and is rejected under the same ground of rejection.

10. Claims 34-35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Little et al. (U.S. 6,907,572 B2) Shalabi et al. (U.S. 2004/0143599 A1) further in view of Gillis et al. (U.S. 6,286,035 B1).

As to claim 34, Little does not teach wherein the parent class further provides a mapping process that allows a specified alias for the identifier. However, Gillis teaches wherein the parent class further provides a mapping process that allows a specified alias for the identifier (unique key; col. 7, lines 56-65 and col. 8, lines 15-20).

As to claim 35, Little does not teach wherein the application comprises a command and the specified alias is provided as an argument to the command when the command is invoked. However, Gillis teaches wherein the application comprises a command (doValidate() routine

with a appropriate argument; col. 7, lines 42-44) and the specified alias is provided as an argument to the command when the command is invoked (unique key of the received parameter; col. 7, lines 60-63).

11. Claims 25, 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gillis et al. (U.S. 6,286,035 B1) in view of Little et al. (U.S. 6,907,572 B2) further in view of Lee (U.S. 6,405,365 B1).

As to claim 25, Gillis does not teach wherein the input source comprises a database table. However, Lee teaches wherein the input source comprises a table (Instructions File 150; col. 8, lines14-15). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Lee to the system of Gillis because Lee teaches data in the database table can be used in the system of Gillis, not only from command line, thus, it would improve the performance of Gillis system.

As to claim 27, Gillis does not teach the input source comprises a script. However, Lee teaches the string comprises a part of a script (next instruction in Instructions File 150; col. 8, lines14-15). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Lee to the system of Little because Lee teaches an alternate method to validate not only a single command but a file of command, thus it would improve the performance of the Little system.

12. Claim 9 is rejected under 35 U.S.C. 103(a) as being unpatentable over Little et al. (U.S. 6,907,572 B2) in view of Shalabi et al. (U.S. 2004/0143599 A1) further in view of Jones (Parse and Validate Command Line Parameters with VB.NET).

As to claim 9, Little and Shalabi do not teach wherein the set of objects comprises .NET objects. However, Jones the set of objects comprises .NET object (In VB.NET, you can obtain the command line passed to the VB.NET via the Command() function; page 2). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Jones to the system of Little and Shalabi because Jones teaches how to parse and validate .NET command line, thus, the system of Little can be improve to validate new type of objects.

13. Claim 24 is rejected under 35 U.S.C. 103(a) as being unpatentable over Gillis et al. (U.S. 6,286,035 B1) in view of Little et al. (U.S. 6,907,572 B2) further in view of Allen (U.S. 6,658,625 B1).

As to claim 24, Little does not teach wherein the input source comprises an XML document. However, Allen teaches a parser parses and validates an XML document (col. 19, lines 35-36). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Allen to the system of Little because Allen teaches a method to parse and validate XML documents, thus the system of Gillis would be improve because it not only can process command but also XML documents.

Response to Arguments

14. Applicant's arguments with respect to claims 1-31 and 33-38 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

15. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. See PTO 892.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to DIEM K. CAO whose telephone number is (571)272-3760. The examiner can normally be reached on Monday - Friday, 7:30AM - 4:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

DC
August 29, 2008

/Diem K Cao/
Acting Examiner of Art Unit 2194